

The following list of claims will replace all prior versions and listings of claims in the application.

#### **LISTING OF CLAIMS**

1. (original) A method for generating a solution to a problem having objects scheduled originally in itineraries, each original itinerary having at least an origin and a destination, the method comprising the steps of: receiving a disruption specification based upon an event, the disruption specification including data identifying the objects to be rescheduled; receiving a request for rescheduling of the objects from a user; grouping the objects to be rescheduled into subproblems, wherein each subproblem is defined by each object therein having the same original origin and destination; applying a first algorithm to each subproblem without allowing varying the origin and destination of the objects in the subproblem for simplification and, in turn, quickly reaching initial solutions; identifying a subclass of objects that are unsuitably rescheduled in the initial solutions; and applying a second algorithm for rescheduling the subclass that allows varying the original itinerary to generate rescheduling solutions for the subclass.

2. (original) A method as recited in claim 1, further comprising the step of applying a third algorithm to an IP problem based upon all of the objects.

3. (original) A method as recited in claim 2, wherein the third algorithm is an IP algorithm with a branch and bound technique.

4. (original) A method as recited in claim 2, further comprising the steps of excluding the subclass of objects from the objects that need to be rescheduled in the disruption specification and applying a fourth algorithm to the remaining objects in the reduced disruption specification to determine rescheduling solutions for the remaining objects.

5. (original) A method according to claim 4, wherein the first and fourth algorithms are transportation simplex algorithms.

6. (original) A method as recited in claim 1, wherein the subclass of objects to be rerouted are identified based upon a suitably of rescheduling criteria.

7. (original) A method as recited in claim 6, wherein identifying the subclass includes determining a cost for each rescheduled object and comparing the cost to a threshold.

8. (original) A method as recited in claim 1, wherein the objects are passengers traveling one or more legs between the origin and the destination.

9. (original) A method as recited in claim 1, wherein the rescheduling solutions include upgrading, downgrading, delaying, and offloading the objects.

10. (original) A method according to claim 1, wherein the second algorithm is selected from the group consisting of the Dijkstra algorithm and a K-shortest path algorithm.

11-31. (cancelled).